33rd Annual International Conference of the IEEE EMBS
Boston, Massachusetts USA, August 30 - September 3, 2011

A Comparison of Non-symmetric Entropy-based Classification trees and Support Vector Machine for Cardiovascular Risk Stratification

Anima Singh and John V. Guttag

Abstract— **Classification tree-based risk stratification models generate easily interpretable classification rules. This feature makes classification tree-based models appealing for use in a clinical setting, provided that they have comparable accuracy to other methods. In this paper, we present and evaluate the performance of a non-symmetric entropy-based classification tree algorithm. The algorithm is designed to accommodate class imbalance found in many medical datasets. We evaluate the performance of this algorithm, and compare it to that of SVM-based classifiers, when applied to 4219 non-ST elevation acute coronary syndrome patients. We generated SVM-based classifiers using three different strategies for handling class imbalance: cost-sensitive SVM learning, synthetic minority oversampling (SMOTE), and random majority undersampling. We used both linear and radial basis kernel-based SVMs. Our classification tree models outperformed SVM-based classifiers generated using each of the three techniques. On average, the classification tree models yielded a 14% improvement in G-score and a 21% improvement in F-score relative to the linear SVM classifiers with the best performance. Similarly, our classification tree models yielded a 12% improvement in G-score and a 21% improvement in the F-score over the best RBF kernel-based SVM classifiers.**

I. INTRODUCTION

Classification tree-based risk stratification models have the advantage of providing easily interpretable classification rules derived from the tree. The classification rules provide justification for why a patient has been classified as high or low risk. This feature makes classification tree-based models appealing for use in a clinical setting, provided that they have comparable accuracy relative to other methods. In this paper, we present a novel non-symmetric entropy-based classification tree algorithm, and compare its performance for cardiovascular risk stratification to those of Support Vector Machines (SVM) based classifiers.

Our algorithm addresses two critical issues in classification tree learning: 1) the order in which variables are selected and 2) discretization of continuous variables. Both of these are made challenging by the class imbalance in medical datasets. Traditional classification tree induction algorithms such as C4.5 [1] and CART [2] use Shannon entropy and the Gini index as measures of class incoherence during tree induction. These measures are suitable under the assumption that both positive and negative examples are well-represented in a dataset. But this is not usually true for medical datasets. For example, amongst the patients who have suffered a mild ACS, only $\approx 2\%$ will suffer cardiovascular death within next three months. In this paper, we present an algorithm that uses a non-symmetric entropy-based class incoherence measure that accounts for the class imbalance in the data.

Another key feature of our algorithm is that it preserves context sensitivity of cutoffs when discretizing continuous variables. For example, it may choose age x as the cutoff for high risk for patients with a particular clinical history and age y as the cutoff for patients with another clinical history. C4.5 and CART also have this feature, but our algorithm uses a distinctive bootstrap aggregating technique [3] for robust estimation of cutoffs.

We test our algorithm on a highly imbalanced medical dataset, consisting of patients who suffered from a non-ST elevation acute coronary syndrome (NSTEACS), i.e., a myocardial infarction without ST-segment elevation or unstable angina. We used cardiovascular death within 90 days as the endpoint. Using the same data set, we generated SVM based classifiers, with linear and radial basis function (RBF) kernels, for risk stratification. To generate SVM classifiers that can handle class imbalance, we used three popular strategies found in the literature: 1) Cost-sensitive SVM learning, 2) Synthetic Minority Oversampling Technique (SMOTE) [4] and 3) Random majority undersampling.

In our experiments, our classification tree models outperformed SVM-based classifiers generated using each of the three techniques. On average, the classification tree models yielded a 14% improvement in G-score [5] and a 21% improvement in F-score [5] over the linear SVM classifiers with the best performance. Similarly, on average our classification tree models yielded at least 12% improvement in G-score and at least 21% in the F-score over the best RBF kernel-based SVM classifiers. All the results were statistically significant.

The rest of the paper is organized as follows. In Section II, we review a commonly used classification algorithm relevant to this application. In Section III, we present an overview of our classification tree algorithm. In Section IV , we present results and analysis of our approach. Finally, Section V presents a summary and conclusions.

This work was supported by Quanta Computer Inc. and Texas Instruments Inc.

Anima Singh is a graduate student in the Department of Electrical Engineering and Computer Science at Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, USA anima@mit.edu

John V. Guttag is a professor in the Department of Electrical Engineering and Computer Science at Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, USA guttag@csail.mit.edu

978-1-4244-4122-8/11/$26.00 ©2011 IEEE

II. BACKGROUND

A. Support Vector Machines

A support vector machine (SVM) [6] is a supervised learning technique that yields a classifier based on a set of N training examples $\{\mathbf{x_1}, \mathbf{x_2}, ...\mathbf{x_N}\}$ and their corresponding class labels $\mathbf{y} \in \{-1, 1\}^N$. Each training example $\mathbf{x_i}$ is an m dimensional vector, representing m features. Given the training examples and their labels, the SVM yields a decision hyperplane that provides a maximum margin separator for the training examples based on their labels. A decision hyperplane is described by a weight vector, $\mathbf{w} = \{w_1, w_2, ..., w_m\}$ and the intercept, b. In the primal form, the SVM for a linear kernel is described as the following optimization problem:

$$min\{\frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N}\xi_i\} \tag{1}$$

subject to the constraints:

$$y_i(\mathbf{w}^T\mathbf{w} + b) \geq 1 - \xi_i; \xi_i \geq 0 \tag{2}$$

where, parameter C penalizes training errors.

For problems where the data from different classes are not linearly separable, a radial basis function (RBF) kernel SVM can be used.

$$K(x_i, x_j) = exp(-\gamma||x_i - x_j||^2) \tag{3}$$

B. Strategies for SVM modeling for Unbalanced Data Classification

The classification performance of an SVM-generated classifier is sensitive to high class imbalance. It is prone to generating a classifier that is biased towards correctly classifying examples from the majority class [7], [8]. We review the traditional approaches used to deal with SVM modeling from unbalanced data. We refer to the examples from the minority class as positive examples.

1) Cost-sensitive SVM: A cost-sensitive SVM (CS-SVM) uses two cost factors C_+ and C_- to adjust the cost of false positive vs. false negatives [9]. The cost-sensitive SVM can be described as the following optimization problem:

$$min\{\frac{1}{2}\mathbf{w}^T\mathbf{w} + C_+\sum_{i:y_i=1}\xi_i + C_-\sum_{j:y_j=-1}\xi_j\} \tag{4}$$

subject to the constraints given by Equation 2.

We choose the cost factors C_+ and C_- such that they satisfy the ratio:

$$\frac{C_+}{C_-} = \frac{\text{number of negative training examples}}{\text{number of positive training examples}} \tag{5}$$

2) Synthetic Minority Oversampling Technique (SMOTE): SMOTE is a oversampling approach in which the minority examples are oversampled by creating 'synthetic' examples [4]. The synthetic examples are created in the feature space. For each minority sample, synthetic samples are introduced along the line segments joining the minority sample with each of the k minority class nearest neighbors.

We use five nearest neighbors in our implementation. After oversampling, the number of positive and negative examples in the training set is roughly equal. We refer to the SVMs generated using SMOTE as SMOTE-SVM.

3) Random Majority Under-sampling: In random majority under-sampling the majority class examples are randomly under-sampled such that the number of positive and negative examples in the training set is roughly equal. We refer to the SVMs generated using random undersampling as RU-SVM.

III. OUR ALGORITHM

In this section, we give a brief description of our binary classification tree induction tree algorithm (Figure 1). The details of the algorithm can be found in [10]. The classification tree is comprised of two phases: the growth phase and the pruning phase.

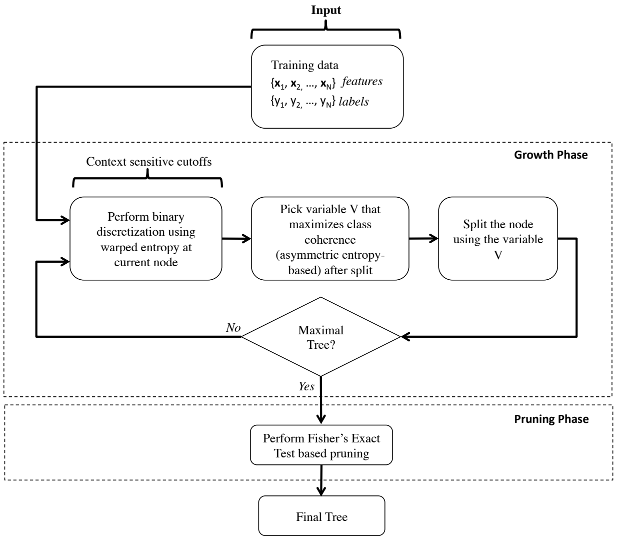
Input
Training data
{x1, x2, ..., xN} features
{y1, y2, ..., yN} labels
Growth Phase
Context sensitive cutoffs
Perform binary discretization using warped entropy at current node
Pick variable V that maximizes class coherence (asymmetric entropy-based) after split
Split the node using the variable V
No
Maximal Tree?
Yes
Pruning Phase
Perform Fisher's Exact Test based pruning
Final Tree

Fig. 1. Algorithm for classification tree learning using non-symmetric entropy measures.

A. The Growth Phase

There are two main steps within the growth phase: 1) binary discretization of continuous variable and, 2) selection of which variable to use to split a given node.

We use a supervised binary discretization method that uses Bagging (Bootstrap aggregating) [11] for robust estimation of cutoffs. The discretization method described in [12] uses warped entropy, a non-symmetric entropy measure. Given a training sample of size N, the discretization method generates r new training sets, called replicates, using Bagging. For each replicate, the method evaluates each candidate cut point using a weighted joint warped entropy (WJE) measure and selects the cutoff that minimizes WJE. The median value of the cutoffs generated from r replicates is the final binary cutoff used to discretize the continuous variable.

Once all the continuous variables are discretized, we select a variable on which to split a given node of the tree. We evaluate each candidate variable based on the class incoherence of the sets generated by splitting the node using that variable. Class incoherence of a set measures the dissimilarity of class label of the examples that belong to the set. The candidate variable that minimizes class incoherence of the resulting split is selected. We use an asymmetric entropy-based class incoherence measure [13], [12].

We repeat the discretization and the variable selection steps until we reach a node with examples that belong to the same class or we have used all the variables along the path from the root node to the given node. We refer to the classification tree generated at the end of the growth phase as the maximal tree.

B. The Pruning Phase

The maximal tree generated in the growth phase is usually large, complex, and over fit to the training data. To improve generalization of the classification tree, we prune the tree using a Fisher's exact test (FET) based pruning approach as described in [14]. We use 0.05 as the p-value threshold for determining whether or not to prune a node of the maximal tree.

After pruning the maximal tree, we assign a specific class label to each leaf node using a weighted majority rule. Examples from the majority class get a weight of 1, while the examples from the minority class get a weight equal to the ratio given in Equation 5.

We refer to the classification tree generated using our algorithm as **N**on-**sym**metric entropy-based **C**lassification **T**ree (NonSym) in the rest of the paper.

IV. EXPERIMENTS

A. Data Set

We used data from 4219 non-ST elevation acute coronary syndrome (NSTEACS) patients, and considered cardiovascular death within 90 days as an endpoint. There were 83 ($\approx 2\%$) cardiovascular deaths within 90 days.

For both NonSym and SVM classifiers, we used as features four continuous variables (age, deceleration capacity [15], heart rate variability (LF-HF) [16] and morphological variability [17] and five discrete variables (history of hypertension, smoking history, prior history of myocardial infarction, history of congestive heart failure, and ST-depression ($\geq$0.5mm)).

B. Methodology

In each of the experiments, we drew 100 training and test sets from the data-set. Each training set contained $2/3$ of the data-set (2813 patients) and its corresponding test set contained the remaining $1/3$ of the data-set (a disjoint set of 1406 patients).

In each of the 100 cases, we used the algorithm described in Section III to induce a classification tree based on the training data set. We evaluated the performance of the classification tree, on the corresponding test set. We also generated and evaluated classification tree risk models using pre-discretized values for the continuous variables (Pre-discretized). The continuous variables were pre-discretized using cut points taken from the literature for cardiovascular risk stratification. The discretized variables were then used to generate classification tree using asymmetric entropy-based class incoherence measure. This experiment was done to investigate the importance of context-sensitive cutoffs for risk stratification. The classification tree algorithm was implemented in MATLAB.

Using the same set of 100 training sets, we generated SVM classifiers using the SVM^{light} package [18]. We picked SVM parameters that yielded the best average classification performance, as measured by G-score and F-score (described below), on the 100 test sets. For the linear SVMs (LSVM), we performed a parameter search for C on an exponentially growing sequence, $C = \{2^{-13}, 2^{-11}, ..., 2^{11}, 2^{13}\}$. For the radial basis kernel-based SVMs (RBFSVM), we performed a grid search for the parameters C and γ on exponentially growing sequences for each. We explored the parameters in the range $\{2^{-13}, 2^{-11}, ..., 2^{11}, 2^{13}\}$.

The risk stratification performance of each classification model was evaluated in terms of G-score (Equation 8), which is the geometric mean of recall and precision, and the F-score (Equation 9) using recall and precision on the minority class:

$$Recall = \frac{\text{true positives}}{\text{true positives + false negatives}} \tag{6}$$

$$Precision = \frac{\text{true positives}}{\text{true positives + false positives}} \tag{7}$$

$$G - score = \sqrt{Recall * Precision} \tag{8}$$

$$F - score = \frac{2 * Recall * Precision}{Recall + Precision} \tag{9}$$

We evaluated the statistical significance of the results using a paired samples t-test [19] and the Wilcoxon test [20]. We consider a difference in performance as statistically significant if the p-value is < 0.05.

C. Results

Table I presents the mean scores for all the different classifiers. NonSym outperformed all the other classifiers. The improvement in both F-score and G-score yielded by NonSym relative to Prediscretized demonstrates the importance of context-sensitive cutoffs for risk stratification.

Table II shows the mean % improvement in performance yielded by our classification tree models relative to the SVM classifiers. NonSym outperformed all SVM classifiers in terms of both G-score and F-score. When compared to the linear SVMs with the best mean scores, NonSym provided a 14% and a 21% mean improvement in G-score and F-score respectively. When compared to the RBF kernel-based SVMs with the best mean scores, NonSym had a 12% and a 21%

TABLE I

MEAN G-SCORE AND F-SCORE OF OUR CLASSIFICATION TREE MODELS COMPARED TO SVM CLASSIFIERS.

	Mean G-score	Mean F-score
NonSym	0.212	0.114
Prediscretized	0.174	0.085
CS-LSVM	0.189	0.094
SMOTE-LSVM	0.190	0.099
RU-LSVM	0.179	0.087
CS-RBFSVM	0.191	0.096
SMOTE-RBFSVM	0.191	0.099
RU-RBFSVM	0.186	0.088

mean improvement in G-score and F-score respectively. All the results were statistically significant.

TABLE II

MEAN % IMPROVEMENT IN G-SCORE AND F-SCORE YIELDED BY OUR CLASSIFICATION TREE MODELS RELATIVE TO THE SVM CLASSIFIERS.

NonSym vs.	**G-score**		
	Mean % Improvement	t-test p-value	Wilcoxon p-value
CS-LSVM	13.6%	<0.001	<0.001
SMOTE-LSVM	14.0%	<0.001	<0.001
RU-LSVM	21.2%	<0.001	<0.001
CS-RBFSVM	12.0%	<0.001	<0.001
SMOTE-RBFSVM	12.4%	< 0.001	< 0.001
RU-RBFSVM	15.4%	<0.001	<0.001

NonSym vs.	**F-score**		
	Mean % Improvement	t-test p-value	Wilcoxon p-value
CS-LSVM	28.1%	<0.001	<0.001
SMOTE-LSVM	21.2%	<0.001	<0.001
RU-LSVM	41.6%	<0.001	<0.001
CS-RBFSVM	25.1%	<0.001	<0.001
SMOTE-RBFSVM	21.2%	<0.001	<0.001
RU-RBFSVM	37.0%	<0.001	<0.001

V. SUMMARY AND CONCLUSIONS

We presented and evaluated a binary classification tree induction algorithm for development of risk stratification models for cardiovascular death. Our algorithm uses non-symmetric entropy based measures for both determining the order of variables in the tree and discretizing continuous variables that are incorporated in the model. The non-symmetric entropy measures allow our algorithm to address the challenge of class imbalance prevalent in medical datasets.

We focus our work on the specific application of risk stratifying patients with ACS. We compared the performance of our classification tree-based models with SVM classifiers. To model SVM from unbalanced data, we generated SVMs using three different approaches: cost-sensitive SVM learning, synthetic minority class under-sampling and random majority class oversampling.

Our results show that in addition to having the advantage of generating interpretable classification rules, our classification tree-based models can also improve risk stratification relative to models generated using SVM. While we demonstrated the utility of non-symmetric entropy-based classification trees only for risk stratification for cardiovascular deaths post-NSTEACS, we believe that it is useful for other clinical applications. However, further investigation and research is required to confirm this hypothesis.

VI. ACKNOWLEDGMENTS

We gratefully acknowledge the financial assistance provided by Quanta Computer Inc. and Texas Instruments Inc.

REFERENCES

[1] J. R. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[2] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[3] T. Qureshi and D. A. Zighed. Using resampling techniques for better quality discretization. In *Proceedings of the 6th International Conference on Machine Learning and Data Mining in Pattern Recognition*, MLDM '09, pages 68–81, Berlin, Heidelberg, 2009. Springer-Verlag.

[4] N.V. Chawla, K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer. SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16(1):321–357, 2002.

[5] Y. Tang, Y.Q. Zhang, N.V. Chawla, and S. Krasser. SVMs modeling for highly imbalanced classification. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 39(1):281–288, February 2009.

[6] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[7] R. Akbani, S. Kwek, and N. Japkowicz. Applying support vector machines to imbalanced datasets. *Machine Learning: ECML 2004*, pages 39–50, 2004.

[8] G. Wu and E.Y. Chang. KBA: kernel boundary alignment considering imbalanced data distribution. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):786–795, June 2005.

[9] K. Morik, P. Brockhausen, and T. Joachims. Combining statistical learning with a knowledge-based approach-a case study in intensive care monitoring. In *Proceedings of 16th ICML*, pages 268–277. Citeseer, 1999.

[10] A. Singh. Risk stratification of cardiovascular patients using a novel classification tree induction algorithm with non-symmetric entropy measures. Master of engineering, Massachusetts Institute of Technology, 2011.

[11] E. Bauer and R. Kohavi. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. In *Machine Learning*, pages 105–139, 1998.

[12] A. Singh, J. Liu, and J. V. Guttag. Discretization of continuous ecg based risk metrics using asymmetric and warped entropy measures. 37:473–476, 2010.

[13] S. Marcellin, D.A. Zighed, and G. Ritschard. Detection of breast cancer using an asymmetric entropy measure. In *Computational Statistics*, volume 25, pages 975–982. Springer, 2006.

[14] W. Liu, S. Chawla, D.A. Cieslak, and N.V. Chawla. A robust decision tree algorithms for imbalanced data sets. In *Proceedings of the Tenth SIAM International Conference on Data Mining*, pages 766–777. Society for Industrial and Applied Mathematics, 2010.

[15] A. Bauer, J.W. Kantelhardt, P Barthel, R. Schneider, T. Mäkikallio, K. Ulm, K. Hnatkova, A. Schömig, H. Huikuri, Bunde A., M. Malik, and G. Schmidt. Deceleration capacity of heart rate as a predictor of mortality after myocardial infarction: cohort study. *The Lancet*, 367(9523):1674–81, May 2006.

[16] M. Malik. Heart rate variability: standards of measurement, physiological interpretation and clinical use. *Circulation*, 93(5):1043–1065, March 1996.

[17] Z. Syed, B. M. Scirica, and S. et. al. Mohanavelu. Relation of death within 90 days of non-st-elevation acute coronary syndromes to variability in electrocardiographic morphology. *American Journal of Cardiology*, 103(3):307–11, Feb 2009.

[18] T. Joachims. Making large-scale support vector machine learning practical. *Advances in kernel methods: support vector learning*, pages 169–184, 1999.

[19] E. Kreyszig. *Introductory Mathematical Statistics*. John Wiley, 1970.

[20] F. Wilcoxon. Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, 1(6):80–83, 1945.